

Project 1 User Manual

1. Executing the Program

This file contains codes for a console-based simulation of a lock in which you can enter a 4-letter combination to trigger related action. The actions of the lock are unlocked and alarm. The lock can accept characters A, B, C, D, E, one at a time. A four-character sequence is needed to crack the lock.

To run this Program, you will need copy of the executable program file, and two table file that are properly formatted.

2. Input Requirements and Restrictions

For the input of “**Enter a letter:**”, input must be one of ‘A’, ‘B’, ‘C’ ‘D’ or ‘E’, all other inputs are treated as invalid.

For the input of “**Try Again(N/Y)?**”, input must be one of ‘n’, ‘N’, ‘y’ or ‘Y’, all other inputs are treated as invalid.

3. Output

3.1 Loading Data

Once the program executes, it will automatically load and setup table files. If fail to open either or those files, user is given unlimited chances to enter a valid file path. For example, if cannot open the transition table file, the console will prompt user:

```
Can't locate the file: "transition_table.txt"
Please re-enter file path:
```

3.2 Main Menu

If data is loaded successfully, the console will first print welcome messages, and ask user for letter input for the lock, the lock can accept letter **A, B, C, D, E**, one at a time.

```
-----
Welcome to the CS 232 Project 1!
This program simulates the a lock when 5 characters are entered.
The actions of the lock are unlock and an alarm.
The lock can accept characters A, B, C, D, E, one at a time.
A four character sequence is needed to "crack" the lock.
-----
Enter a letter:
```

Error Checking for letter input:

User is given unlimited chances to re-enter if the letter input is invalid.

For instance, if input is **O**, then the console will prompt user:

```
Enter a letter: O
Please enter one of A B C D E as your input, try again.
Enter a letter:
```

If input is **ABCDEFGEGHDVW**, then the console will prompt user:

```
Enter a letter: ABCDEFGEGHDVW
Invalid input, try again: |
```

3.3 Trigger Lock Action:

If the user enters a 4-letter combination that doesn't crack the lock, user would be prompted:

```
Enter a letter: A
Enter a letter: A
Enter a letter: A
Enter a letter: A
*** Lock Action: Alarming ***
```

If the user enters a 4-letter combination that cracks the lock, user would be prompted:

```
Enter a letter: D
Enter a letter: E
Enter a letter: A
Enter a letter: D
*** Lock Action: Door Unlocked! ***
```

3.4 Re-crack:

After a lock action is triggered, user is given unlimited chances to re-crack the lock.

If the input is 'n' or 'N', then this commands lead to farewell message and terminate the program:

```
Try Again?(Y/N) N  
Have a nice day!
```

If the input is 'y' or 'Y', user is given a new round to crack the lock:

```
Try Again?(Y/N) Y  
Enter a letter:
```

Error Checking for re-crack input:

All the other inputs are invalid, and user would be prompted:

```
Try Again?(Y/N) D  
Please enter a 'N' or 'Y'
```